MetaMap Source Distribution README

Willie Rogers

September 4, 2009

Contents

1 Prerequisites

Platforms The MetaMap Datafile Builder currently only runs on the Linux and Solaris 9 or greater.

Sun Java Runtime Environment (JRE) 1.5 or later Sun's Java 1.5.0 or later is not required but is preferred for use of the MetaMap Source Code Suite. JRE 1.4 or later is required. Java is available from the "Developer Resources for Java Technology" website (http://java.sun.com/).

Quintus Prolog 3.5 available from Swedish Institute of Computer Science (SICS), the Quintus Prolog website: http://www.sics.se/quintus/

GNU C Compiler should already be installed or be available as a package in your distribution if you're using Linux. If you're using Solaris you can get it from sunfreeware.com (http://www.sunfreeware.com). See Solaris specific prerequisites for more information.

GNU Binutils should already be installed or be available as a package in your distribution if you're using Linux. If you're using Solaris you can get it from sunfreeware.com (http://www.sunfreeware.com). See next item "Solaris specific prerequisites" for more information.

GNU Make should already be installed or be available as a package in your distribution if you're using Linux. If you're using Solaris you can get it from sunfreeware.com (http://www.sunfreeware.com). See next item "Solaris specific prerequisites" for more information.

Berkeley DB 4.5 should already be installed or be available as a package in your distribution if you're using Linux. Otherwise, you can get it from http://www.oracle.com/database/berkeley-db/index.html. The Berkeley DB Release Page is at: http://www.oracle.com/technology/software/products/berkeley-db/db/index.html. MetaMap is written to use the Berkeley DB 4.5 libraries. It is not expected to run properly using another other version of Berkeley DB, including later versions.

Solaris specific prerequisites See file README_dfb_solaris.html (online version: http://metamap.nlm.nih.gov/README_dfb_solaris.html) for more information on Solaris specific prerequisites.

2 Getting the MetaMap Distribution

The public MetaMap distribution can be downloaded at the Download Section of the MetaMap website:

http://metamap.nlm.nih.gov/#Downloads

The latest version of this document can be downloaded at:

HTML http://metamap.nlm.nih.gov/README_src.html

PDF http://metamap.nlm.nih.gov/README_src.pdf

3 Extracting the distribution

Use the following `tar` command extract the distribution in the same directory where the Public MetaMap distribution was extracted:

```
$ bzip2 -dc public_mm_linux_2009.tar.bz2 | tar xvf -
$ bzip2 -dc public_mm_src_2009.tar.bz2 | tar xvf -
```

`Tar` will create the distribution directory `public_mm`. Note: The data compression program BZIP2 (available from http://www.bzip.org) is required to decompress the distributions. GNU tar is preferred, but not required to extract the contents of the distributions.

4 Installing the distribution

Connect to the new directory created by extracting the distribution and invoke the install program:

```
$ cd <distribution directory>
$ ./bin/install.sh
```

A sample run of the installation script follows:

```
$ ./bin/install.sh
Enter basedir of installation [/home/wrogers/Projects/public_mm_dist/test/linux/public_mm]
```

```
Basedir is set to /home/wrogers/Projects/public_mm_dist/test/linux/public_mm.

The WSD Server requires Sun's Java Runtime Environment (JRE)
Sun's Java Developer Kit (JDK) will work as well. if the
command: "which" java returns /usr/local/jre1.4.2/bin/java, then the
JRE resides in /usr/local/jre1.4.2/.

Where does your distribution of Sun\'s JRE reside?
Enter home path of JRE (JDK) [/usr/local/jdk1.6.0_11]:

Using /usr/local/jdk1.6.0_11 for JAVA_HOME.

/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/WSD_Server/config/disambServer.cfg g
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/WSD_Server/config/log4j.properties g
Setting up bin directory scripts:
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/make_all generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/metamap09 generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/SKRenv generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/SKRmake generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/skrmedpostctl generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/SKRrun generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/uninstall.sh generated.
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/wsdserverctl generated.
Setting up test suite:
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/TestSuite/buildTestSuite09.sh genera
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/TestSuite/runTest_2009.sh generated.
Checking for required datafiles
Checking for optional datafiles (WSD)
Public MetaMap Install complete.
Running MetaMap source development environment setup...

MetaMap Source Install

Public MetaMap Basedir: /home/wrogers/Projects/public_mm_dist/test/linux/public_mm

Where does your Quintus Prolog installation reside?
Enter base dir of Quintus Prolog installation [/nfsvol/crfiler-ind/II_Research/quintus]:

Where does your GNU C compiler reside?
Enter base dir of GNU C Compiler [/usr]:

Where does your Berkeley DB 4.5 installation reside?
Enter base dir of Berkeley DB 4.5 installation [/usr]:
/usr/local/BerkeleyDB.4.5-m32
Berkeley DB 4.5 sharable library found.
libdb.so sharable library found.
Setting up bin directory SKRenv script:
```

```
/home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin/SKRenv generated.

Source Install Settings:

QUINTUS basedir: /nfsvol/crfiler-ind/II_Research/quintus
GCC basedir: /usr
GCC compiler: /usr/bin/gcc
Berkeley DB basedir: /usr/local/BerkeleyDB.4.5-m32
Public MetaMap Source Install complete.

Public MetaMap Install Settings:

Public MetaMap basedir: /home/wrogers/Projects/public_mm_dist/test/linux/public_mm
Public MetaMap Program Dir: /home/wrogers/Projects/public_mm_dist/test/linux/public_mm/bin
Java Home dir: /usr/local/jdk1.6.0_11

$
```

5 Instructions for using compiling MetaMap

Add the $BASEDIR/public_mm/bin directory to your program path, where BASEDIR is where you placed the MetaMap public distribution/

To do this in tcsh or csh do the following:

```
$ set path = ( ${BASEDIR}/public_mm/bin $path )
```

Similarly, use the following command in bash or sh:

```
PATH=${BASEDIR}/public_mm/bin:$PATH
export PATH
```

Then move to the $BASEDIR/public_mm directory and type:

```
$ cd $BASEDIR/public_mm
$ make_all

... various messages and warnings ...

No compilation errors.
```

If there are no errors, then go to the $BASEDIR/public_mm/src/skr and type:

```
$ cd src/skr
$ SKRmake

... various messages and warnings ...
```

```
### Linking OK

*************************************************************
     The a.out binary has been deposited in
     /home/wrogers/Projects/public_mm_dist/test/linux/public_mm/src/skr
*************************************************************
```

6 Running the compiled metamap executable

You can run the compiled executable from the $BASEDIR/public_mm/src/skr use the command:

```
$ SKRrun -M /DATA/XDR -B /BDB4 ./a.out <metamap arguments>
```

For example:

```
$ SKRrun -M /DATA/XDR -B /BDB4 ./a.out

MetaMap (2009)


Control options:
Berkeley DB databases (normal strict 09 model) are open.
Static variants will come from table varsan in /home/wrogers/Projects/public_mm_dist/test/linux
Derivational Variants: Adj/noun ONLY.
Accessing lexicon /home/wrogers/Projects/public_mm_dist/test/linux/public_mm/lexicon/data/BDB4/
Variant generation mode: static.

|: obstructive sleep apnea
|:
Established connection to Tagger Server on localhost.
Processing 00000000.tx.1: obstructive sleep apnea

Phrase: "obstructive sleep apnea"
Meta Candidates (11):
  1000 Obstructive sleep apnoea (Sleep Apnea, Obstructive) [Disease or Syndrome]
   901 Apnea, Sleep (Sleep Apnea Syndromes) [Disease or Syndrome]
   827 APNOEA (Apnea) [Pathologic Function]
   827 Sleep [Organism Function]
   827 Obstructive (Obstructed) [Functional Concept]
   827 Apnea (Apnea Adverse Event) [Finding]
   793 Sleeping (Asleep) [Finding]
   755 Obstruction [Individual Behavior,Pathologic Function]
   755 Sleepy [Finding]
   755 Sleeplessness [Sign or Symptom]
   755 Obstruction (Obstruction within Medical Device) [Phenomenon or Process]
Meta Mapping (1000):
  1000 Obstructive sleep apnoea (Sleep Apnea, Obstructive) [Disease or Syndrome]
```

```
|: heart attack and lung disease
|:
Established connection to Tagger Server on localhost.
Processing 00000000.tx.1: heart attack and lung disease

Phrase: "heart attack"
Meta Candidates (8):
  1000 Heart attack (Myocardial Infarction) [Disease or Syndrome]
   861 Heart [Body Part, Organ, or Organ Component]
   861 Attack, NOS (Onset of illness) [Finding]
   861 Attack (Attack device) [Medical Device]
   861 attack (Attack behavior) [Social Behavior]
   861 Heart (Entire heart) [Body Part, Organ, or Organ Component]
   861 Attack (Observation of attack) [Finding]
   827 Attacked (Assault) [Injury or Poisoning]
Meta Mapping (1000):
  1000 Heart attack (Myocardial Infarction) [Disease or Syndrome]

Phrase: "and"

Phrase: "lung disease"
Meta Candidates (11):
  1000 Lung Disease (Lung diseases) [Disease or Syndrome]
   972 pulmonary disease (Reported previous pulmonary disease) [Finding]
   972 Pulmonary Disease (Physicians - Pulmonary Disease) [Professional or Occupational Group]
   861 Disease [Disease or Syndrome]
   805 MAL (MAL gene) [Gene or Genome]
   805 MAL (MKL1 gene) [Gene or Genome]
   805 MAL (TIRAP gene) [Gene or Genome]
   805 MALS [Finding]
   694 Lung [Body Part, Organ, or Organ Component]
   694 Lung (Entire lung) [Body Part, Organ, or Organ Component]
   601 Pneumonia [Disease or Syndrome]
Meta Mapping (1000):
  1000 Lung Disease (Lung diseases) [Disease or Syndrome]
|: ^D
$
```